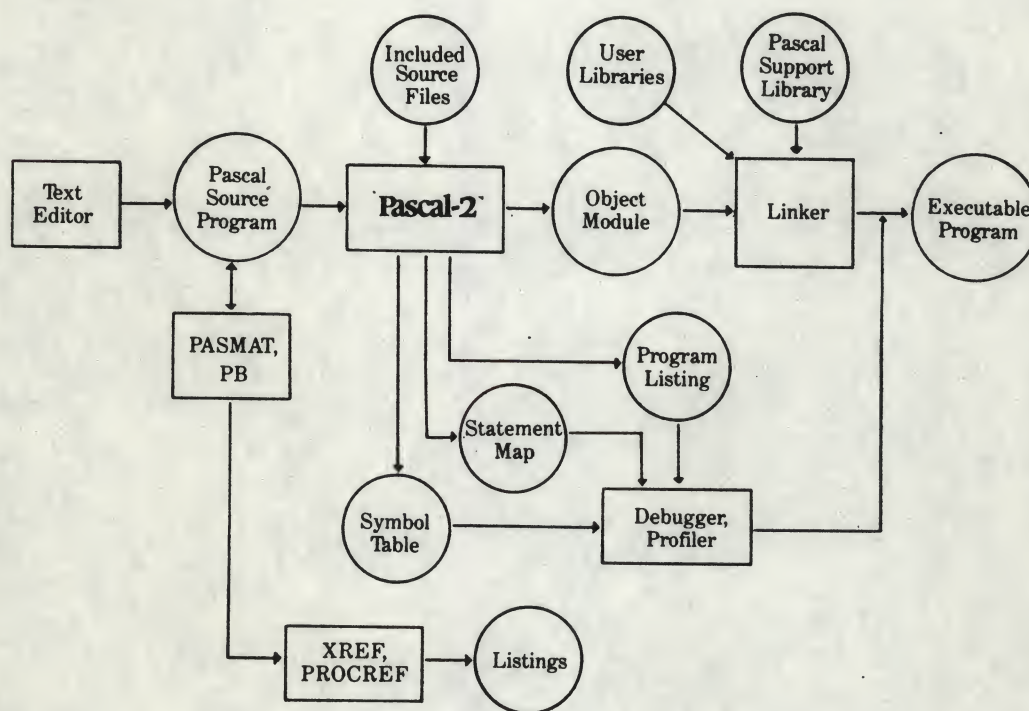


## Pascal-2 V2.1/RT-11 Introduction

Pascal-2 is an integrated system for software development. At the heart of the system is a transportable multipass compiler that adheres to the Pascal standard while performing optimizations to generate compact, fast code. The Pascal-2 system also offers sophisticated error checking during compilations, extensive error reporting and recovery at run-time, a Debugger to examine the dynamic state of a running program in a high-level Pascal context, plus other development utilities. Together, these components offer the professional programmer a structured and unified environment in which to design, code, test, maintain, and improve software. The result should be the production of more reliable programs in less time than with other programming packages. Further, use of the Pascal-2 compiler will allow programs to be transported to other computer systems with a minimum of change, thereby accelerating future software development.

Developed over several years, Pascal-2 grew out of our experience with our first Pascal compiler, Pascal-1. Pascal-1 is a one-pass compiler specific to Digital Equipment Corporation's PDP-11 series, with low-level extensions giving the programmer control over the PDP-11 hardware and operating system. Pascal-2 is larger and compiles more slowly than Pascal-1, but Pascal-2 produces code that is much smaller and faster than Pascal-1 code. Typical programs are 30 to 40 percent smaller and up to twice as fast.



**The Pascal-2 Software Development System**

*The Pascal-2 system consists of the Pascal-2 compiler, the support library, the formatters PASMAT and PB, the Debugger and Profiler, and the cross-references XREF and PROCREF. The text formatter PROSE, not shown, is also a component of the system. The user creates the Pascal source program, the included source files, and user libraries. The Text Editor and Linker are supplied by the computer vendor.*



1890

1. The first part of the paper is devoted to a general discussion of the subject. It is shown that the problem is of great importance and that it has not been fully solved. The author then proceeds to a detailed examination of the various methods which have been proposed for its solution. It is found that each of these methods has its own merits and its own defects. The author then proposes a new method which he believes to be superior to all the others. This method is based on the principle of the conservation of energy and it is shown that it is capable of solving the problem in a much more efficient manner than any of the other methods.

2. The second part of the paper is devoted to a detailed examination of the various methods which have been proposed for its solution. It is found that each of these methods has its own merits and its own defects. The author then proposes a new method which he believes to be superior to all the others. This method is based on the principle of the conservation of energy and it is shown that it is capable of solving the problem in a much more efficient manner than any of the other methods.

3. The third part of the paper is devoted to a detailed examination of the various methods which have been proposed for its solution. It is found that each of these methods has its own merits and its own defects. The author then proposes a new method which he believes to be superior to all the others. This method is based on the principle of the conservation of energy and it is shown that it is capable of solving the problem in a much more efficient manner than any of the other methods.



## Pascal-2 V2.1/RT-11 Introduction

### About Version 2.1

Version 2.1 of the Pascal-2 software represents the first major technical improvement in the Pascal-2 software since its release in 1981. The product now includes conformant array parameters, raising the compiler to Level 1 of the proposed international standard. The implementation of lazy I/O changes the way Pascal-2 performs input operations, where input is not read until it is actually needed. Numerous compiler and support library bugs have been fixed. Run-time error diagnostics now include a procedure-by-procedure walkback from the point of error to the main program and are capable of trapping I/O errors, allowing users to recover from I/O errors with their own code. Users may change the wording of run-time error messages and print additional information about the error for debugging purposes.

Enhancements of the Pascal-2 software are summarized in the following list; detailed explanations of each feature appear in the appropriate sections of the manual.

#### Summary of Changes, New Features in V2.1 Software

<u>Feature</u>	<u>Function</u>
Lazy I/O	An I/O scheme in which data is not read until actually used in the program.
Conformant Array Parameters	Allows you to write general procedures that accept array parameters of different size and with different lower and upper bounds.
Run-Time Errors	Additions include new error numbers and messages, and explanations of those messages. User processing of run-time errors gives you control of run-time error reporting. You may change the wording of run-time error messages and print additional information about the error. The run-time error walkback aids in diagnosis of run-time errors, pinpointing the location of the error in Pascal source terms and showing the reverse sequence of procedure calls leading to the error.
'Nowalkback' Switch	Disables the error walkback. Available as either compilation or embedded switch. This switch is "off" by default (the walkback is printed).
I/O Error Trapping	You can now recover from I/O errors with your own code.
'Workspace:n' Switch	Reduces/increases the work space required by the compiler to compile a program.
Support Library	Library entry points now have the form P\$ <i>nnn</i> , where <i>nnn</i> is an integer in the range 0..135. Most routines are now called in the same way as other Pascal procedures, using the standard Pascal calling sequence.
Predefined Procedures	New procedures <code>getpos</code> and <code>setpos</code> simulate random access to files of type <code>text</code> . Function <code>space</code> determines the amount of stack and heap available to an executing program. New procedures <code>rename</code> and <code>delete</code> allow you to rename or delete files from within a program.
Files	Files opened local to a procedure are now closed upon procedure exit. Also, the <code>WK:</code> option allows you to specify the device to which the compiler directs its working data files.
Eight-Bit Characters	Allows you to use extended character sets.







## Summary of Changes, New Features in V2.1 Software (cont.)

<u>Feature</u>	<u>Function</u>
Non-Decimal Integer Constants	Allows the use of integers in radices ranging from base 2 (binary) to base 16 (hexadecimal).
Size Restrictions	Easing of certain restrictions allows larger programs to be compiled.
Expanded Unsigned Integers and Functions	The unsigned integer range is 0..65535. Unsigned functions are capable of returning unsigned and structured values.
String Package	Includes the new routines <code>equal</code> , <code>assign</code> and <code>assignchar</code> . Implemented with conformant array parameters. Procedure <code>delete</code> is now named <code>deletestring</code> , to prevent conflict with the new predefined procedure <code>delete</code> .
'%Include' Directive	New syntax allows specification of the disk volume number of the included file.

Because the code generated by the V2.1 compiler differs from that of V2.0, current Pascal-2 users must recompile existing Pascal-2 programs and external modules with the new compiler. Some programs may need to be changed in order to compile or redesigned to take advantage of features such as conformant array parameters, lazy I/O and user control of run-time error reporting.

## Pascal-2 Documentation Package

The Pascal-2 user documentation contains information on the use of the Pascal-2 compiler and related utilities on Digital's RT-11 operating systems: RT-11 V4 and V5, SJ, XM, and TSX-Plus. In general, we assume that readers of the manual are programmers familiar with Pascal and the RT-11 operating system. Some sections assume a detailed working knowledge of the language.

The *Pascal-2 User Manual* is not intended to be a Pascal textbook. Beginners can make their way carefully through this manual, but we refer you to the reading list in the appendix, "For More Information."

The manual consists of five major guides, as follows:

- The User's Guide serves as a quick overview of the Pascal-2 system, to give you a feel for how it works. The guide, written on a beginner's level, takes you through the basic steps of compiling, correcting, and running a Pascal-2 program. The User's Guide also has brief explanations and examples of some of the standard features and utilities of the Pascal-2 system.
- The Programmer's Guide contains detailed descriptions of compilation commands, embedded and low-level switches, and the low-level interface between Pascal-2 and the operating system. The Programmer's Guide also contains a miscellany of information on implementation-related problems, divided into two broad categories: error recovery and implementation notes. Finally, the guide describes Pascal-2's optimizations and provides helpful hints as to the cause of compile-time and run-time errors and ways to fix the errors.



THE UNIVERSITY OF CHICAGO

DEPARTMENT OF CHEMISTRY

RESEARCH REPORT

ON THE

STRUCTURE OF

THE

CRYSTAL

OF

THE

SUBSTANCE

OF

THE

CRYSTAL

OF

THE

CRYSTAL

OF

THE

CRYSTAL

OF

THE

CRYSTAL

OF

THE

CRYSTAL

OF

THE

CRYSTAL

OF



- The Language Specification describes Pascal-2's language features in detail. Since the second edition of Jensen and Wirth's *User Manual and Report* in 1978, the language has undergone major changes, which are incorporated in the Third ISO Draft Proposal 7185. Because not everyone is familiar with that document, the Language Specification begins by summarizing those changes and describing the ways that Pascal-2 implements them. Thus, the guide serves not only as a description of our Pascal product but also as a review of the language's evolution since 1978.
- The Debugger and Profiler Guide describes two programs designed to alleviate tedious aspects of programming or to improve the usefulness of the Pascal-2 system. The Debugger helps find and correct errors that cannot be caught at compile time. The execution Profiler shows areas of the program in which the most time is spent.
- The Utilities Guide describes each of the following packages: program formatters, a text formatter, cross-reference programs, a package that helps interface assembler routines with Pascal-2 programs, and a dynamic string package. Each utility is described in detail, with examples.

Included with the manual is a set of release notes including the Installation Guide.

For information on the RT-11 system, see these RT-11 manuals: *Introduction to RT-11*, *System User's Guide*, *Software Support Manual*, *System Utilities Guide*.

In addition, Pascal-1 customers upgrading to Pascal-2 should refer to the *Pascal-2 Conversion Guide* and the CONVRS utility, which are available from Oregon Software. The *Conversion Guide* explains specific language differences between Pascal-1 and Pascal-2 and the practical programming problems created by the differences. The guide describes the use of the CONVRS utility to help isolate areas in a Pascal-1 program that will have to be modified to convert to Pascal-2; the guide then details the steps required to convert the programs. The *Conversion Guide* concludes with a list of solutions to errors that you may encounter while completing the conversion to Pascal-2.

## Style Notes

The *Pascal-2 User Manual* follows these style conventions:

### Text:

Pascal reserved words, predefined symbols, switches and compiler directives are in boldface typewriter type: **begin**, **write**, **%include**, **nomain**. Portions of examples referred to in text are in boldface typewriter type. System directives are in upper-case boldface typewriter type: **.SETTOP**, **.CALFIP**. Program and system names are in upper case: **ROTAT**, **RT-11**.

### Program Examples:

Commands that you should type are in underlined boldface typewriter: **RUN EX**. These commands assume a carriage return at the end.

### Program Listings:

The Pascal-2 compiler accepts any combination of upper-case and lower-case characters. Examples in this manual have Pascal words in lower case and have user-defined words with an initial capital letter and other capitalization as needed for readability, as shown in this program fragment:

```
procedure Show;
begin
    SomeUserAction;
    writeln(Result);
end;
```







Single quotes ('..') in examples and in text appear as '..'.

**Terminology:**

We use standard terms as they are used in documents describing the RT-11 operating system.

**Support Policy**

The license fee for your Pascal-2 system includes one year of software support, which covers the following:

- Telephone assistance. We'll provide a quick cure to your problem if at all possible.
- Formal, written response to all problems, suggestions, and comments received in writing. For complex problems, we need written descriptions of your technical problem to ensure correct diagnosis and repair. (This service does **not** include applications consultation.)
- A no-cost update to the latest revision of the Pascal-2 system, upon the written request of your Designated Contact Person. This is the standard response to bugs that have been fixed. (We do charge for the media used for the update.)
- The Oregon Software Pascal Newsletter, which contains status reports on all of our Pascal products, announcements of new versions of software and new products, and various technical articles.

Support may be renewed annually. Customers of an Oregon Software distributor should contact their distributor for support.



1. The first part of the paper is devoted to a general discussion of the problem of the origin of life. It is shown that the problem is one of the most important and most difficult in the history of science.

2. The second part of the paper is devoted to a detailed discussion of the various theories of the origin of life. It is shown that the most plausible theory is that of the spontaneous generation of life from non-living matter. This theory is supported by the fact that life is found in the most hostile environments on Earth, and that life is found in the most hostile environments in the universe.

3. The third part of the paper is devoted to a discussion of the various experiments that have been conducted to test the various theories of the origin of life. It is shown that the most convincing experiments are those that have shown the spontaneous generation of life from non-living matter. These experiments are supported by the fact that life is found in the most hostile environments on Earth, and that life is found in the most hostile environments in the universe.